AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application.

LISTING OF CLAIMS:

1-2. (Canceled)

3. (Currently amended) A cache subsystem, comprising: a cache controller; and a data memory coupled to the cache controller, the data memory holds a contiguous block of memory defined by an address stored in a register accessible to the cache controller; wherein the data memory is adapted to store two groups of local variables, a first group comprising local variables associated with finished methods and a second group comprising local variables associated with unfinished methods; wherein, based on a threshold value, local variables associated with the second group are fetched from memory that is external to the cache subsystem; and ~~The cache subsystem of claim 1~~ wherein the threshold value is determined from allocation bits associated with each of a plurality of lines comprising the data memory.

4. (Currently amended) A cache subsystem, comprising: a cache controller; and a data memory coupled to the cache controller, the data memory holds a contiguous block of memory defined by an address stored in a register accessible to the cache controller; wherein the data memory is adapted to store two groups of local variables, a first group comprising local variables associated with finished methods and a second group comprising local variables associated with unfinished methods; wherein, based on a threshold value comprising an address that points to a line in the data memory that

separates the first and second groups, local variables associated with the second group are fetched from memory that is external to the cache subsystem; and ~~The cache subsystem of claim 2~~ wherein a reference to local variables is compared to said threshold address and a local variable from external memory is fetched if said reference is to an invalid line and is below said threshold address and a local variable is not fetched from external memory if said reference is above said threshold address.

5. (Currently amended) The cache subsystem of claim 42 wherein methods are invoked and each time a method is invoked, the threshold address is adjusted.

6. (Original) The cache subsystem of claim 5 wherein the threshold address also is adjusted when a method finishes.

7. (Currently amended) A cache subsystem, comprising: a cache controller; and a data memory coupled to the cache controller, the data memory holds a contiguous block of memory defined by an address stored in a register accessible to the cache controller; wherein the data memory is adapted to store two groups of local variables, a first group comprising local variables associated with finished methods and a second group comprising local variables associated with unfinished methods; wherein, based on a threshold value, local variables associated with the second group are fetched from memory that is external to the cache subsystem; and ~~The cache subsystem of claim 1~~ wherein said data memory comprises a plurality of lines and said value comprises an allocation bit associated with each line.

8. (Original) The cache subsystem of claim 7 wherein each line also includes a valid bit and local variables are fetched from memory depending on the valid and allocation bits.

9. (Original) The cache subsystem of claim 7 wherein the allocation bits are set to one value for local variables associated with finished methods and to another value for local variables associated with unfinished methods.

10-11. (Canceled)

12. (Currently amended) A processor adapted to couple to external memory, comprising: a processing core on which a plurality of methods execute; and a cache subsystem accessible to the processing core and comprising a data memory coupled to a cache controller, the data memory holds a contiguous block of memory defined by an address stored in a register accessible to the cache controller; wherein the data memory is adapted to store a first group of local variables associated with finished methods and a second group of local variables associated with unfinished methods; wherein, based on a threshold value comprising an address that points to a line in the data memory that separates the first and second groups, local variables associated with the second group are fetched from memory that is external to the cache subsystem; and ~~The processor of claim 11~~ wherein a reference to local variables is compared to said threshold address and a local variable from external memory is fetched if said reference is to an invalid line and is below said threshold address and a local variable is not fetched from external memory if said reference is above said threshold address.

13. (Currently amended) The processor of claim 12~~1~~ wherein methods are invoked and each time a method is invoked, the threshold address is adjusted.

14. (Original) The processor of claim 13 wherein the threshold address also is adjusted when a method finishes.

15. (Currently amended) A processor adapted to couple to external memory, comprising: a processing core on which a plurality of methods execute; and a cache subsystem accessible to the processing core and comprising a data memory coupled to a cache controller, the data memory holds a contiguous block of memory defined by an address stored in a register accessible to the cache controller; wherein the data memory is adapted to store a first group of local variables associated with finished methods and a second group of local variables associated with unfinished methods; and wherein, based on a threshold value, local variables associated with the second group are fetched from memory that is external to the cache subsystem; and ~~The processor of claim 10~~ wherein said data memory comprises a plurality of lines and said value comprises an allocation bit associated with each line.

16. (Original) The processor of claim 15 wherein each line also includes a valid bit and local variables are fetched from memory depending on the valid and allocation bits.

17. (Original) The processor of claim 15 wherein the allocation bits are set to one value for local variables associated with finished methods and to another value for local variables associated with unfinished methods.

18. (Currently amended) A processor adapted to couple to external memory, comprising: a processing core on which a plurality of methods execute; and a cache subsystem accessible to the processing core and comprising a data memory coupled to a cache controller, the data memory holds a contiguous block of memory defined by an address stored in a register accessible to the cache controller; wherein the data memory is adapted to store a first group of local variables associated with finished methods and a second group of local variables associated with unfinished methods; and wherein, based on a threshold value, local variables associated with the second group are fetched from

memory that is external to the cache subsystem; and ~~The processor of claim 10~~ wherein the data memory comprises a plurality of lines, each line having an allocation bit and an associated state, and the threshold value is determined from the state of the allocation bits.

19. (Previously presented) A method, comprising: allocating space in a data memory in which variables are stored that are used by methods; setting a value indicative of which variables are used by finished methods and which variables are used by unfinished methods; and wherein, based on the value, fetching local variables associated with the second group from external memory that is separate from the data memory.

20. (Original) The method of claim 19 wherein setting the value includes adjusting a pointer address.

21. (Original) The method of claim 19 wherein setting the value includes setting a plurality of allocation bits associated with a plurality of lines provided in the data memory.

22. (Currently amended) The cache subsystem of claim 7~~1~~ wherein ~~only~~ local variables associated with the second group are fetched from memory that is external to the cache subsystem.

23. (Currently amended) The processor of claim 15~~0~~ wherein ~~only~~ local variables associated with the second group are fetched from memory that is external to the cache subsystem.

24. (Currently amended) The method of claim 19 wherein ~~only~~ local variables associated with the second group are fetched from memory that is separate from the data memory.

25. (Currently amended) The cache subsystem of claim ~~7~~1 wherein local variables associated with the second group are fetched from memory that is external to the cache subsystem or saved to the external memory.

26. (Currently amended) The processor of claim 1~~5~~0 wherein local variables associated with the second group are fetched from memory that is external to the cache subsystem or saved to the external memory.

27. (Previously presented) The method of claim 19 wherein local variables associated with the second group are fetched from external memory that is separate from the data memory or are saved to the external memory.

28. (Previously presented) A method comprising the steps of:

using a limit to optimize accesses between a first memory and an external memory;

deciding which local variables need to be saved into the external memory when preparing for a context change;

deciding after the context change if a miss needs to lead to a line fetch in external memory.